69th International Astronautical Congress, Bremen, Germany. Copyright © 2018 by California Institute of Technology.

IAC–18–IAC-18,C1,8,12,x45307

Do You See What I See? Interactive Visualization of Mission Design and Navigation

Jeffrey Stuart
Jet Propulsion Laboratory, California Institute of Technology, USA, Jeffrey.R.Stuart@jpl.nasa.gov

Basak Ramaswamy
Jet Propulsion Laboratory, California Institute of Technology, USA,
Emine.Basak.Alper.Ramaswamy@jpl.nasa.gov

Try Lam
Jet Propulsion Laboratory, California Institute of Technology, USA, Try.Lam@jpl.nasa.gov

Nathaniel Guy
Jet Propulsion Laboratory, California Institute of Technology, USA, Nathaniel.Guy@jpl.nasa.gov

Frank Laipert
Jet Propulsion Laboratory, California Institute of Technology, USA, Frank.E.Laipert@jpl.nasa.gov

Alex Menzies
Jet Propulsion Laboratory, California Institute of Technology, USA, Alexander.Menzies@jpl.nasa.gov

Nicholas Bradley
Jet Propulsion Laboratory, California Institute of Technology, USA, Nicholas.E.Bradley@jpl.nasa.gov

Aprameya Mysore
Jet Propulsion Laboratory, California Institute of Technology, USA, Aprameya.Mysore@jpl.nasa.gov

Nitin Arora
Jet Propulsion Laboratory, California Institute of Technology, USA, Nitin.Arora@jpl.nasa.gov

Mission Design and Navigation (MDNav) is an intensive process requiring advanced computational resources, expert human intuition, and many successive human-in-the-loop iterations to converge on acceptable trajectory designs or navigation solutions. The current bottleneck in MDNav is not the underlying computational algorithms but the human cognitive capacity to prune through a multitude of simulated results to select high-value candidates. One approach to alleviate this burden is through the judicious application of visualizations that allow humans to interactively filter data in multiple dimensions to reveal salient patterns and highlight divergences. When designed efficiently, such interactive visualizations should aid human operators to get familiar with data faster, visually observe correlations, and communicate findings more effortlessly. In this work, we present three visualization case studies that have the potential to increase human operator efficiency in MDNav. While identifying the most critical "pain points" that operators face, and also working on potential solutions, we followed a human-centered design approach. We started with a series of interviews with potential users, and then rapidly created prototypes for alternative solutions, validated outcomes with feedback from users through out development of these proof of concept visualizations. With this survey of our current efforts, we demonstrate the transformative capability of interactive data visualizations for improving mission development and operations, enabling operators to grow intuition, and communicating key concepts across diverse mission teams.

I. Introduction

Mission design and navigation (MDNav) is a computationally intensive field incorporating a variety of modeling and simulation techniques as well as sources of information. From initial feasibility studies to in-flight operations, operators are constantly engaged with producing, curating, analyzing, and communicating large amounts of data, both simulated and from real-world measurements. The current constriction within MDNav is not computational power, however, but the analytic abilities of the human-in-the-loop analysts tasked with understanding and effectively utilizing this data to inform mission development and the continued operation of on-orbit assets. Thus, any steps taken to improve human efficiency and reduce the time-to-insight can have a disproportionate impact on project success. Though there are many avenues to increase operator capability, in this investigation we focus on the implementation of interactive visual environments for rapid iteration of simulations and intuitive processing of complex data sets.

While representing astrodynamic data in visual format has a long and storied history extending prior to the beginning of the Space Age, only comparatively recently have computer speed and graphical capabilities advanced to the point that human-in-the-loop, real time interaction is quick enough for every day use. Initial concept studies of immersive visualizations were conducted in the 2000's,[1,2] though most development to date has focused on desktop (i.e., flat screen) use cases. For example, Systems Tool Kit (STK),[3] FreeFlyer,[4] the General Mission Analysis Tool (GMAT),[6] and Copernicus[7] offer extensive design and operations capabilities but are primarily list and menu driven in their interaction. Cosmographia offers a more integrated experience visual but relies on user-supplied SPICE trajectory files to operate.[5] On the other end of the scale, the Monte library is almost entirely script and command line driven but offers extensive ability to conduct customized astrodynamic analysis and has a visualization suite (SparQ) that is relatively easy for users to extend.[8,9]

In more recent years, concerted efforts have been made to apply elements of visual analytics to the process of trajectory design, particularly within the context of multibody dynamics. Schlei and others have applied interactive Poincaré maps as well as dynamic clipping of trajectory segments to create end-to-end initial guesses for orbit transfers.[10–12] Likewise, interactive catalogs of periodic and quasi-periodic orbits have been created that enable users to browse through an interactive visual environment to analyze potential mission orbits.[13,14] Extensions of Poincaré mapping techniques have also pushed interactivity into multi-dimensional spaces.[15,16] Relatively fewer efforts have been made outside of direct trajectory design applications, however the Europa Clipper mission has recently applied visual techniques to assess how well mission requirements are met during design iterations.[17] Attempts have also been made to provide more interaction when visualizing large data sets, for example space debris and small solar system bodies.[18] Outside of astrodynamics, an extensive literature on trajectory visualization, data representation, and human-computer interfaces may provide relevant techniques for MDNav analysis; relevant examples for spacecraft trajectories include visualizations of complex aeronautic[19] and ground-based[20] paths.

In this investigation, MDNav specialists have partnered with design experts to assess the potential for interactive visualization across the breadth of mission design and navigation. While traditional trajectory analysis challenges will be addressed, our goal is to widen the scope of visualization to areas that have seen comparatively limited development and address the most critical "paint points" of the MDNav work flow. In particular, we seek to create a multi-mission library of visualization suites targeted toward specific types of analyses; rather than an "all-in-one" interface, we wish to give mission teams the flexibility to define their own custom environments as needed for their specific mission analysis. We begin by discussing the full scope of MDNav within the life cycle of a space mission, then discuss relevant details of interactivity and the rapid prototyping development model integrated into our effort. Three case studies, with prototyped solutions, are presented in mission design, Monte Carlo analysis, and orbit determination and initial observations and lessons learned are discussed.

II. Mission Life Cycles

The different phases of a spaceflight mission[21] entail a wide variety of MDNav tasks ranging in complexity, computational needs, dynamical fidelity, and work flow. Each broadly defined task category requires the use of specialized analysis tools; when feasible, generalized computational libraries may be used to provide a common backbone, but unique enhancements are almost always required for deep space missions. Similarly, a spectrum of visualization techniques are available for human users to interact with simulation software and interpret results, with some elements being common to multiple tasks. In Table 1, we provide an initial assessment of the scope for visualization within MDNav with an eye toward identifying key areas of focus for the development of interactive visual analysis environments. In the table, check marks (✓) with green backgrounds are used to designate intersections where a particular task clearly requires or is greatly enhanced by a specific visual-

Table 1: Mission life cycle phases, analysis tasks, and visualization needs

Interactive Visualization Needs	**Mission phases**						
	Concept Formulation (Phases Pre-A-B)		**Implementation** (Phases C-D)		**Operations** (Phases E-F)		
	Guess & Check	Broad Search	Targeting & Optimization	Statistical Analyses	Orbit Determination	Maneuver Planning	Sequencing & Verification
Trajectory Viewer	✓	∼	✓	∼	∼	∼	✓
Inputs & Models	∼	✓	∼	✓	✓	∼	∼
Timeline Viewer	∼	∼	✓	∼	✓	✓	✓
Astrodynamic Plots	✓	∼	∼	∼	✓	✓	✓
Data Clouds		✓		✓	∼	∼	
Constraints	✓	∼	✓		∼	∼	∼
Iterations	✓		✓		✓	✓	
Raw Images					✓		∼

ization capability; the "approximately" symbol (∼) indicates an area where a visualization could potentially be helpful or may only occasionally be needed. The task categories indicated by grey backgrounds (*Guess & Check*, *Statistical Analyses*, and *Orbit Determination*) are the motivating cases for the current investigation.

In this investigation, we define the various MDNav tasks across a project life cycle as follows:

- *Physical Feasibility* - initial operating principals are established and concept verified to not violate known physical laws (not shown in table)
- *Guess & Check*[*] - initial trade space exploration looking for feasible options of any level of quality; heavily influenced by human intuition
- *Broad Search* - systematic survey across significant portion of the trade space; often a massively parallel search for global optimum via machine learning or combinatorial optimization algorithms
- *Targeting & Optimization* - localized convergence and optimization of a single baseline trajectory satisfying all mission constraints
- *Statistical Analyses* - assess robustness of baseline solution to disturbances and errors, including ΔV / propellant budgeting, planetary protection, and upset recovery; usually Monte Carlo simulations, though other uncertainty quantification techniques can be used
- *Orbit Determination* - collect measurements (including optical navigation), filter into updated estimates of spacecraft state, maneuver execution, environment, etc.
- *Maneuver Planning* - design of deterministic and statistical maneuvers for execution by the spacecraft; for orbit maintenance and/or targeting future events (e.g., flybys)
- *Sequencing & Verification* - translation of maneuvers and observations into command sequences for on-board execution; checks against desired behavior and known limitations

Note that *Physical Feasibility* is not included in the table or in our subsequent discussion because this stage is does not (usually) require numerical simulations and so computer-based visualization is irrelevant. In general, analysis tasks progress from initial searches for trajectory solutions to detailed optimization and refinement to in-flight navigation, though back-and-forth iteration is common over a mission life cycle and the overlap between work assignments and analysis techniques is often significant. Once a spacecraft is in flight, the three **Operations** tasks are iterated in relatively rapid succession for the remainder of the mission lifetime; in the case of extended missions, elements of the "pre-launch" tasks are often conducted again to reach new celestial targets or science orbits.

We also identify several areas for visualization within the context of MDNav:

- *Trajectory Viewer* - three-dimensional or projected views of the spacecraft trajectory, potentially overlaid with additional indicators for accessory information (e.g., markers for control points, view cones for instruments)
- *Inputs & Models* - structured representation of inputs to simulation and models used, particu-

[*]Also known as the Terrestrially Regulated Iterative Algorithm for Learning and Estimation of Realistically Recognizable Operative Representations (TRIAL&ERROR)[22]

larly useful if multiple cases can be compared / considered simultaneously

- *Timeline Viewer* - graphical indication of sequence of events with overlaid information like maneuvers, flybys, tracking intervals, ground analysis, etc.

- *Astrodynamic Plots* - interactive versions of commonly used plots in mission design and navigation (e.g., B-plane targeting, measurement residuals, time histories, coverage plots)

- *Data Clouds* - visualization of large sets of physical and/or abstract data, including state uncertainties and other statistical phenomena; common techniques are scatter plots, volume rendering, and histograms

- *Constraints* - visual indication of constraint violations and/or satisfaction of mission requirements

- *Iterations* - automatic tracking and quick recall of design iterations / steps in numerical corrections process; intuitive summarization of "nearness" of solutions across sequence

- *Raw Images* - processing and analysis of raw images from on-board cameras, used for optical navigation

While we have detailed each visualization separately, there is clearly room for overlap and combination between many of these needs. Indeed, our chief goal is to explore links between these visualization approaches and find effective ways to integrate them into a larger interactive analysis environment. For example, *Constraints* violations could be viewed in a stand-alone figure but they can also be highlighted within a *Trajectory Viewer* or *Timeline Viewer*; likewise, by linking visualizations of *Iterations* and *Constraints*, a user could rapidly identify any trouble spots during the course of an optimization run. A relevant example from the literature is Schlei's linking of Poincaré maps (an instance of *Data Cloud* scatter plot visualization) to a *Trajectory Viewer* that enables a user to rapidly piece together orbit segments into an end-to-end trajectory.[10–12]

III. Interactive Visualization Methodology

1

III.i Interactive Data Visualizations

Data visualizations refer to graphical representations that map data dimensions to various visual attributes such as location, color, size and shape. While this mapping is more straightforward for spatial data sets representing a physical entity, such as a trajectory, the mapping from non-spatial data dimensions to visual attributes requires careful design considerations.

The goal of visualization is to amplify human cognition,[23] thus effective visualization designs take into account how human-perceptual system processes visual signals, how these signals are interpreted to formulate insights, as well as what questions are valuable to answer about data. Visualizations work by revealing similarity relationships between data points or data categories in terms of visual attribute similarities. By recognizing shared properties between data points or groups, users can gain insights about patterns and correlations hidden in the data, as well as identify outliers.

There are a plethora of visualization techniques ranging from simple scatter plots and bar charts, to more complex visualizations like adjacency matrices and parallel coordinates.[24] Each visualization technique has strengths that emphasize certain properties in the data. For instance, if the user is interested in understanding the distribution of data on two specific dimensions, a scatter plot would be ideal, which places data points that are similar in value, closer in screen coordinates. If the user wants to understand the distribution of data on many dimensions at the same time, a parallel coordinates visualization would be more appropriate. Hence, picking the right visualization technique for the right job requires understanding how users reason about their data sets.

III.ii Interaction

The real power of data visualizations emanates from the interactions that allow users to formulate and refine visual queries as they gain new insights from the data. The interactions either allow users to modify how visual attributes are mapped, or simply support data operations like filtering and sorting. One of the key concepts in information visualization is called Multiple Coordinated Views where users can view different dimensions of data visualized in different representations side by side. The coordination happens when the user selects a group of data points on one display, automatically they are highlighted on others. This helps users to formulate insights like, "data points that are similar in attribute x, are also similar in their attribute y".

As users observe trends and identify anomalies in the data, they constantly gain new insights, and reformulate their data queries. This continuous filtering, sorting and immediate visual feedback paradigm afforded by interactive visualizations allows users to focus on data without the interruption of going back to a script or data analysis tool to formally type a new query for each possible data representation.

III.iii Human-Centered Design Approach

Many data sets that users have to make sense of are very high dimensional and structurally rich. For instance, users might need to make sense of multiple time-series spatial data in the context of a static model described as a hierarchy of non-spatial attributes, as in the Orbit Determination case discussed below.

Since data sets are rich, and ways to visualize them are endless, designing effective visualizations requires understanding how users, who are experts of one domain, reason about their data, what types of questions they need to answer, and in what order. Overcoming this challenge requires collaboration between visualization design experts and domain experts. Human-centered design provides techniques and processes to guide the collaboration between designers of visualization tools and domain experts.

While there are many different techniques and approaches that can be employed, the process of human-centered design can be summarized as focusing on users early in the exploration process, synthesizing input from users into actionable insights, turning ideas quickly into tangible prototypes, and iterating over designs with users in the loop.

In the case studies described below, we followed a human-centered design approach, starting with open-ended interviews to identify possible application areas where interactive visualizations can aid domain experts in MDNav. Note that not every analysis task requires human attention. On the contrary, the majority of data analysis tasks are trivial and quickly automated. However, in certain areas, humans still need to review alternative solutions, interpret outcomes with expertise, and pick options with intuition. Later, we continued with more focused interviews in selected problem areas. Next, we developed prototypes and iterated over design alternatives with feedback from designated domain experts. When possible we conducted focus group studies, showcasing the prototypes, to gather feedback from larger groups of experts.

IV. Interactive Visualization Case Studies

In this initial investigation, we have identified key "pain points" related to *Guess & Check*, *Statistical Analyses*, and *Orbit Determination* and seek to address them as follows:

- *Guess & Check* - we approach initial design challenges in interplanetary transfers and spacecraft formation design by integrating *Trajectory Viewers*, *Constraints*, *Iterations*, and *Astrodynamic Plots*
- *Statistical Analyses* - we parse a large set of outputs from a electric propulsion missed thrust Monte Carlo analysis by creating interlinked *Data Cloud* and *Timeline Viewer* windows
- *Orbit Determination* - we connect an interactive *Inputs & Models* tree with *Astrodynamic Plots* to intuitively compare multiple navigation cases

In all cases, our goal is to provide experienced operators with visual environments that greatly increase their efficiency and speed at completing specific tasks within an MDNav work flow. However, the prototype tools developed have also proven to be excellent tools for communicating key concepts and challenges within team settings, particularly when non-MDNav experts are present.

IV.i Trajectory Design in 2D and 3D

Our first goal is to develop visual environments suitable for the rapid design of feasible trajectories in preliminary concept studies or refinement of existing baseline trajectories as mission requirements are defined. Along these lines, we have created: 1) a prototype VR tool for interacting with and altering a patched conic interplanetary trajectory; and, 2) an extension of the Monte/SparQ software suite that enables rapid manipulation of the relative orbits forming a space-based interferometric observatory. These test cases highlight ways in which *Trajectory Viewers*, *Constraints*, *Iterations*, and *Astrodynamic Plots* can be interlinked to enable rapid exploration of a complex trajectory design trade space.

Beginning with the interplanetary VR, we seek to provide a trajectory analyst an immersive experience where a trajectory can be viewed and directly manipulated in all three spatial dimensions. Figure 1 is a snapshot of this VR package, nicknamed "Slingshot", developed using HTC Vive hardware and Unity software[25] and highlighting key features, including the HTC control wand interacting with one of the trajectory control points. As the user moves the con-

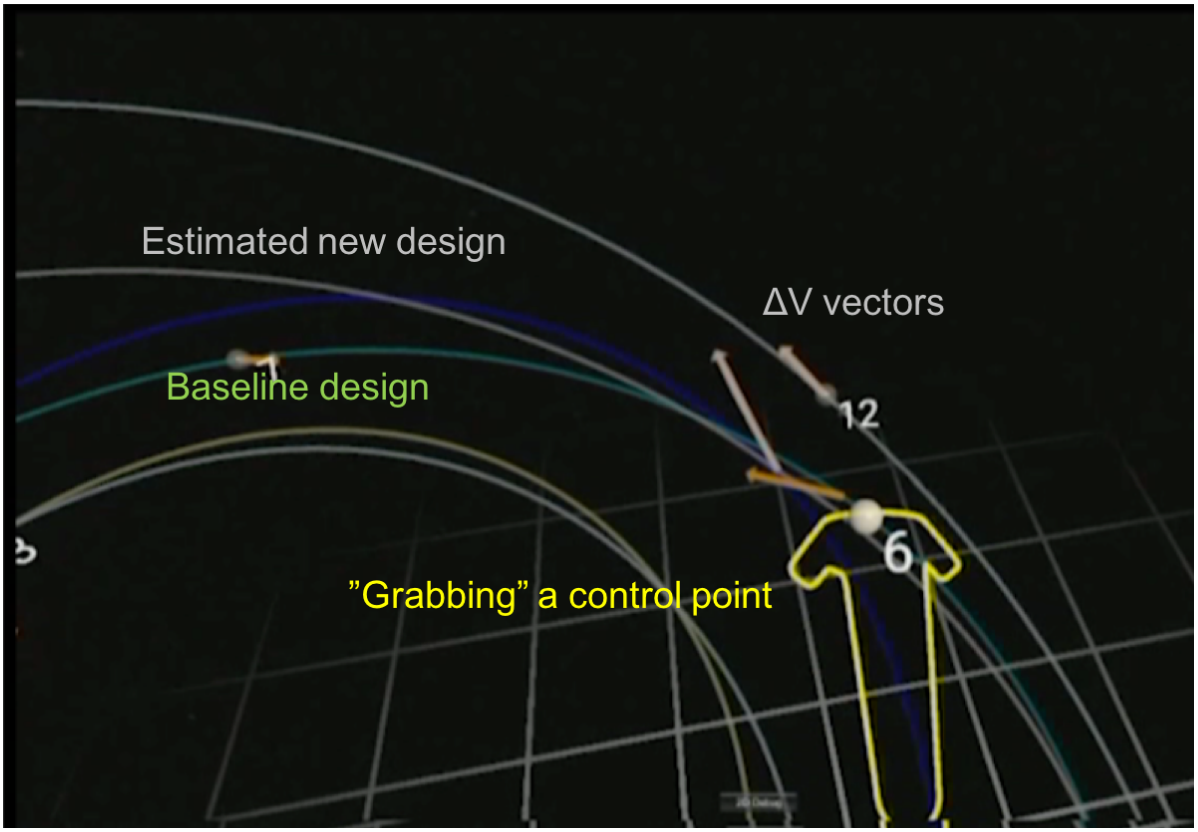


Fig. 1: Virtual reality view of interplanetary transfer, moving control points in 3D automatically updates patched conic trajectory (HTC Vive / Unity)

trol point in 3D space, Lambert arcs[26] connecting the nodes are automatically computed and displayed to the user as grey segments previewing the updated trajectory; when the user releases control of the point, the trajectory is updated to the new design. In addition to illustrating the path of the spacecraft within the solar system, Slingshot highlights the ΔV vectors associated with maneuvers performed at the control points, giving the user an intuitive sense of changes in propellant cost and thrusting direction. The seamless connection of 3D view and 3D manipulation captures the true spatial extent of the trajectory in ways that are difficult if not impossible using typical computer monitors and peripherals. The grid in the background simultaneously acts as a measure of distance (much like graph paper or a grid on a plot) and serves to mark the floor of the physical room, alleviating disorientation while the user is within the VR scene. The controller can also toggle an animation of the spacecraft following the current trajectory design, with accessory information like epoch and distance from the sun displayed within easy view of the user. This immersive VR environment, while powerful for an individual user, also promises to be especially beneficial in concurrent engineering sessions when a diverse team of experts can query the spacecraft trajectory for their own specialized needs (e.g., solar power, thermal, radiation, telecommunications) and offer real-time feedback to the MDNav expert.

Building upon the VR example, we have also created a prototype augmented reality demonstration for the Europa Clipper mission using the Microsoft Hololens, shown in Fig. 2. To highlight the effectiveness of the human-centered design approach, we note that key features of this visualization were driven by user feedback, for example the Clipper mission need for a more collaborative implementation than the single-user VR suite. Likewise, the Clipper team has a mature set of design and navigation software but is currently working to demonstrate that mission requirements are being met by the trajectory design. This AR prototype enables the user to quickly move between different phases of the Europa flyby campaign, illustrating the resulting surface coverage for

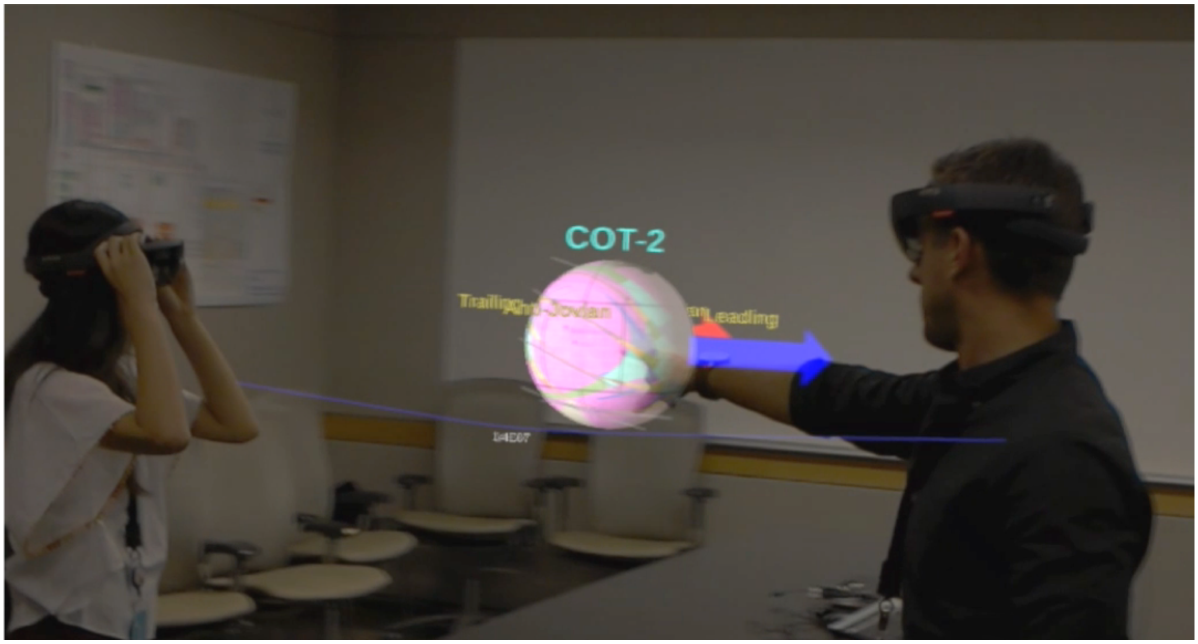


Fig. 2: Augmented reality view of Europa with Europa Clipper flyby trajectories and instrument coverage maps (Microsoft Hololens)

different instruments to groups of team members or reviewers. Likewise, additional features like key direction arrows and labels were incorporated based on user feedback and the rapid iteration cycle.

As another example of visually interactive trajectory design, we create a custom extension of the Monte astrodynamic computation library using the SparQ astrodynamic visualization suite[8,9] as well as PyQt.[27] As a test case, we use the design of the relative orbits of a formation flying cluster of spacecraft composing a radio interferometer.[28] We implement as design variables both the classic Clohessy-Wiltshire (CW) equations[29] as well as more recent mathematical formulations based upon linear combinations of invariant flow structures;[28,30] note that, for the circular orbit case considered here, the two sets of variables are mathematically equivalent, though they do provide different insights into the overall design problem.[31] A portion of the design environment, dubbed "KelvinBall", showing automatically updating views of the relative trajectories and PyQt widgets for design of the relative orbits is shown in Fig. 3. Completing the work space are three matplotlib windows,[33] displayed in Fig. 4, showing the absolute magnitude of the spacecraft/spacecraft separations, a 3D view of the interferometric baselines formed by each pair of spacecraft, and a plot detailed how the interferometric requirements are met across time. The visual environment links all these front-end displays to an interactive Monte session as a back-end computational engine such that updates are simulated on the fly without requiring explicit action from the user, i.e., there is no "recompute" button, though the Monte command line remains available for further flexibility.

Each element of KelvinBall plays a key role within the formation analysis work flow and highlights desirable features for user interactivity. Beginning with the portions in Fig. 3, the SparQ window enables quick switching of the frame and central body for the trajectory views, as well as a time slider setting the current epoch and defining the timespan of the analysis. Phasing angle control dials for all six spacecraft are presented in the main design panel (built using PyQt widgets), and push buttons for each spacecraft pop-up and additional design panel with controls for the remaining CW parameters; when one of these dials is adjusted, the new design values are fed to the Monte back-end and the trajectories are automatically recomputed. The design parameters are grouped in this way because the phasing angles are the most likely to be adjusted and often need to be adjusted for individual spacecraft in rapid succession; in contrast, the remaining parameters more rarely used, but when comprehensive trajectory changes are required, all parameters are needed simultaneously.

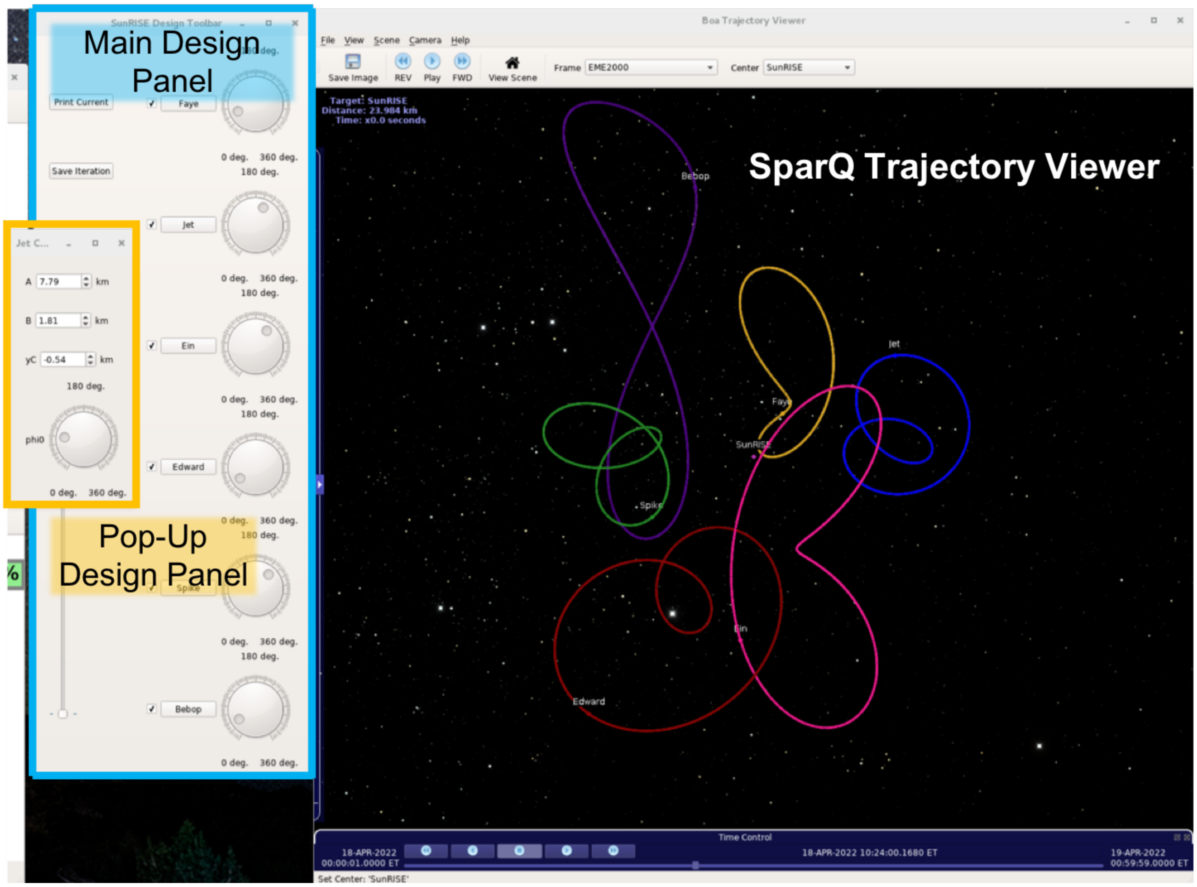
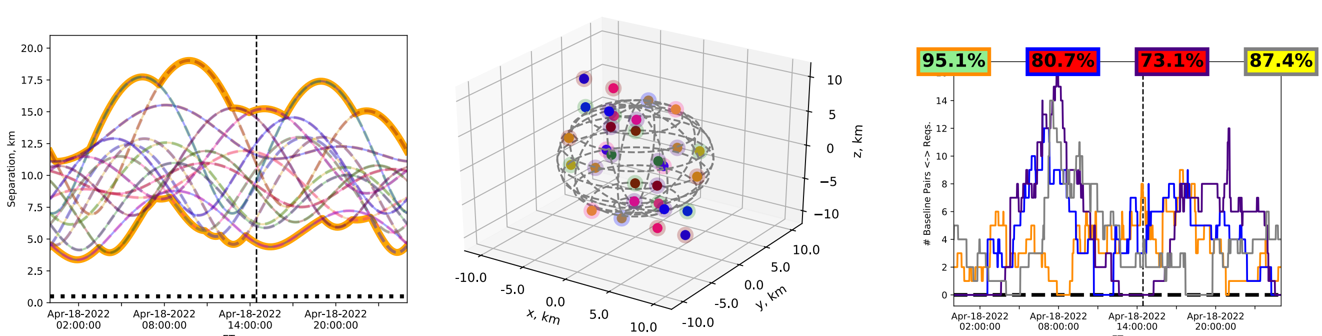


Fig. 3: Visual design environment for spacecraft formation, showing PyQt dials and spin boxes for Clohessy-Wiltshire parameters (Monte/SparQ)



(a) sc/sc separations (b) interferometer baselines (c) mission requirements

Fig. 4: Spacecraft formation accessory plots, automatically update as relative orbits are changed inside the visual design environment (Python matplotlib)

Also in the primary panel are check boxes to selectively suppress an individual spacecraft, a useful feature for contingency and resiliency assessments. Behind the pop-up panel in Fig. 3 is a set of buttons and a slider for saving and recalling designs during iteration; this feature enables the operator to explore freely without fear of losing a high-quality solution.

Turning now to the elements in Fig. 4, the three accessory plots are automatically updated as the formation design is changed via the PyQt widgets. In Fig. 4(a), the separations between individual spacecraft pairs are charted as dashed lines alternating the corresponding colors in the trajectory views in Fig. 3; the orange highlights on the top and bottom emphasize the closest approaches and furthest recessions for the formation as a whole. The instantaneous spacecraft/spacecraft baselines forming the observatory are presented in a 3D plot, Fig. 4(b), allowing the user to rotate and view the interferometer from any vantage point and therefore assess the performance relative to a variety of targets. Finally, time histories of the interferometric performance are shown in Fig. 4(c), where the plot shows how the mission requirements are met over the course of one orbit. Each line represents a different phase within the multi-month mission, with corresponding summary boxes at the top of the figure. The summary boxes give the percentage of the orbit that mission requirements are met and the background color changes to give the user a quick and intuitive sense of the performance (green is good, yellow is marginal, and red is unsatisfactory).[†] The left and right subfigures also have vertical dashed lines indicating the current epoch, linked to the epoch slider in the SparQ window. This allows the user to quickly identify and move to epochs of interest, honing in on trouble spots and quickly resolving formation design issues. While rigorous studies have yet to be conducted, anecdotal use of KelvinBall has demonstrated orders of magnitude increases in operator efficiency over command line runs generating static plots; design alterations that used to take hours at a minimum can now be conducted in a matter of minutes.

In addition to the formation design environment, we have also branched off the AR development for Europa Clipper to create a Hololens visualization, seen in Fig. 5, of the formation flying interferomenter. Our goal with this visual prototype is to interactively illustrate how the relative motion of the spacecraft forms the baselines of the interferometer and thus satisfies the mission requirements over time. In the snapshot taken, the full 3D baselines between each spacecraft pair are being formed prior to being projected onto the 2D interferometer plane. The interactive environment also permits a user to isolate the baselines emanating from a single spacecraft out toward the rest of the formation, enabling rapid assessment of the relative contribution of each spacecraft. The AR environment also enables multiple members of the mission and science team to work together within the same scene, quickly proposing and testing hypotheses related to the interferometer performance. As with the rest of the design prototypes, this AR demonstration greatly eases the burden of communication across the team and significantly reduces the time to insight.

IV.ii Monte Carlo Analysis of Missed Thrust

Our second MDNav task, *Statistical Analyses*, is a critical part of mission development and safety, from developing margined propellant budgets to ensuring planetary protection requirements are met. As a motivating example, we consider missed thrust analysis for an Earth-Mars transfer enabled by a low-thrust solar electric propulsion system.[34] Since the spacecraft is thrusting nearly continuously, any unexpected engine outage (e.g., safe mode) can significantly impact the transfer duration and propellant cost. Thus, a Monte Carlo analysis is conducted wherein stochastic outages are enforced upon a baseline trajectory, prompting a re-optimization of the interplanetary transfer; note that multiple missed thrust intervals can occur per mission. In this investigation, 2000 Monte Carlo cases are run to provide a representative sampling of potential operational profiles.

To enable mission designers to interrogate the resulting collection of disparate solutions, we prototype a web-based work space built using D3.js[35] and combining scatter plots, histograms, and timeline viewers, as illustrated in Figs. 6 and 7. All panes within the environment are dynamically linked together, as showcased in the upper left of Fig. 6: the grey region in the upper left scatter plot corresponds to the zoomed area in the middle scatter. Both scatters plot arrival delay in days along the x-axis and additional propellant consumed in the y-axis, and the colors of the dots indicate the number of outages for that Monte Carlo run. Also in the middle scatter, a mouse hover on one of the individual samples provides a brief summary of that transfer, including the lateness in arrival at Mars, the propellant margin, and the total outage duration for the transfer. Note

[†]We have intentionally degraded the formation design to highlight this feature of KelvinBall; the true baseline design meets mission requirements across all phases.

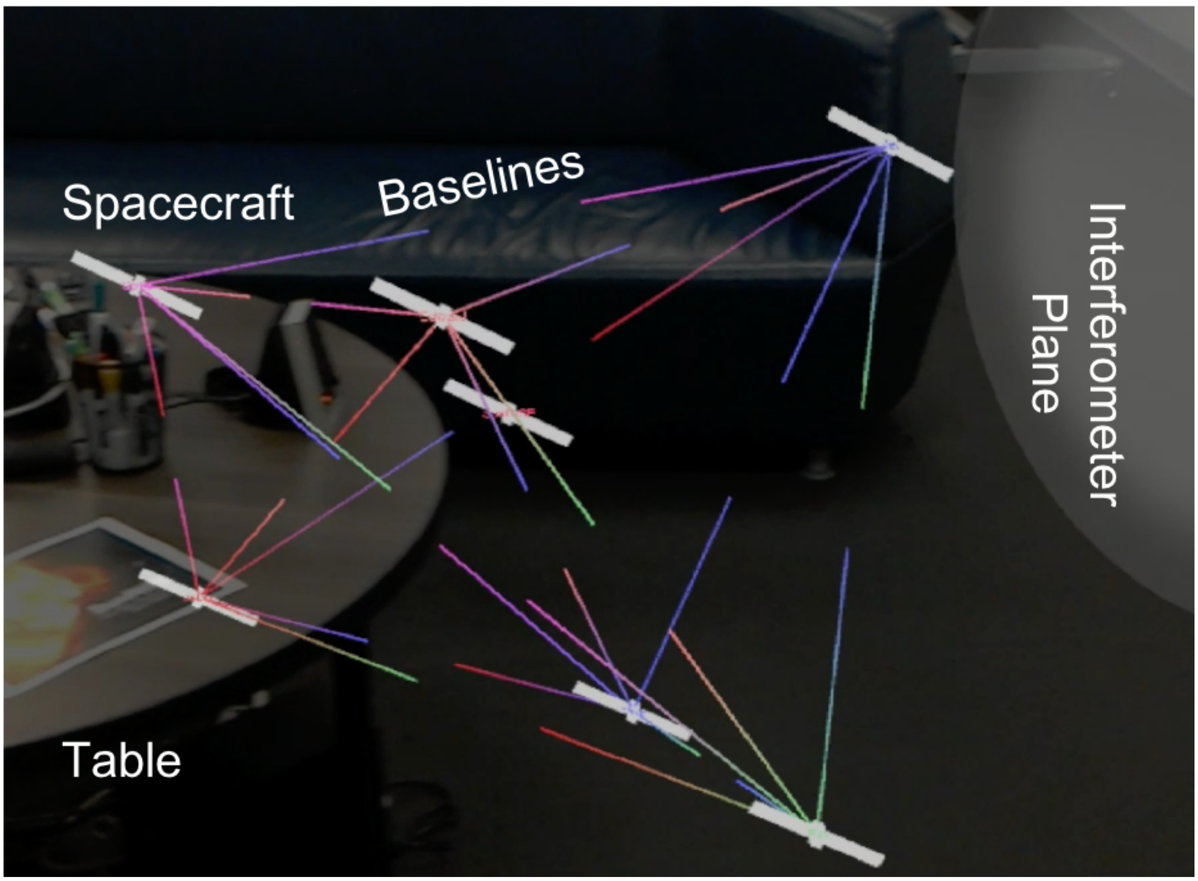


Fig. 5: Augmented reality view of space-based interferometer, showing relative spacecraft-spacecraft baselines (Microsoft Hololens)

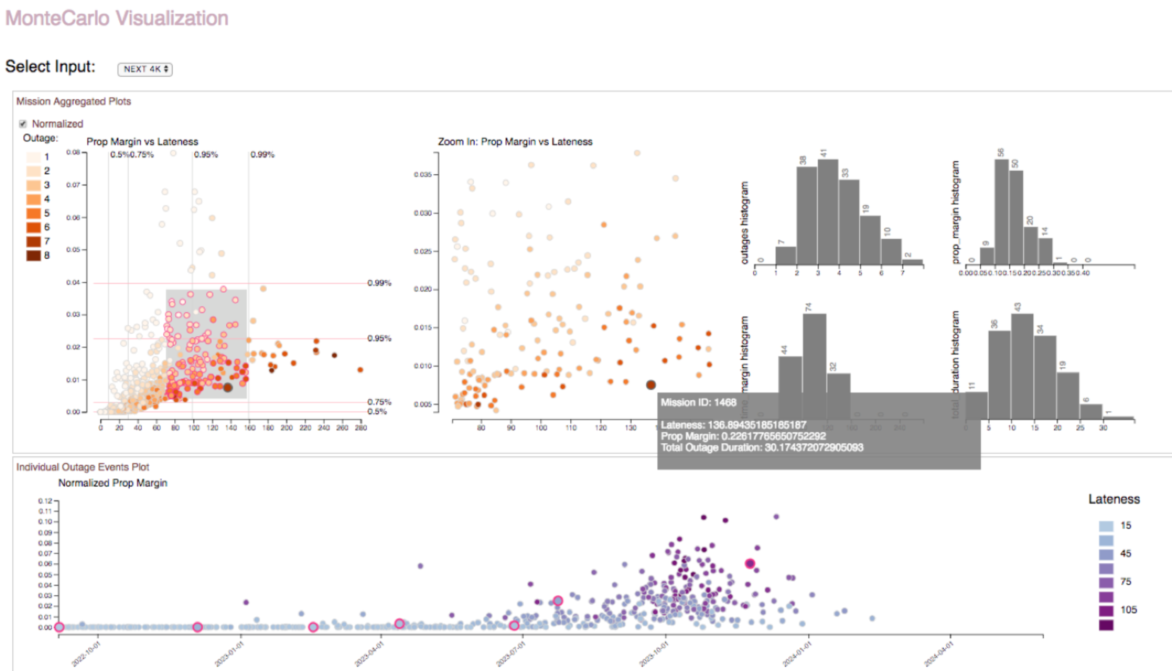


Fig. 6: Missed thrust Monte Carlo result interrogation environment, showing time delay vs. propellant margin scatters, histograms, and time line of outages for multiple samples (D3.js)

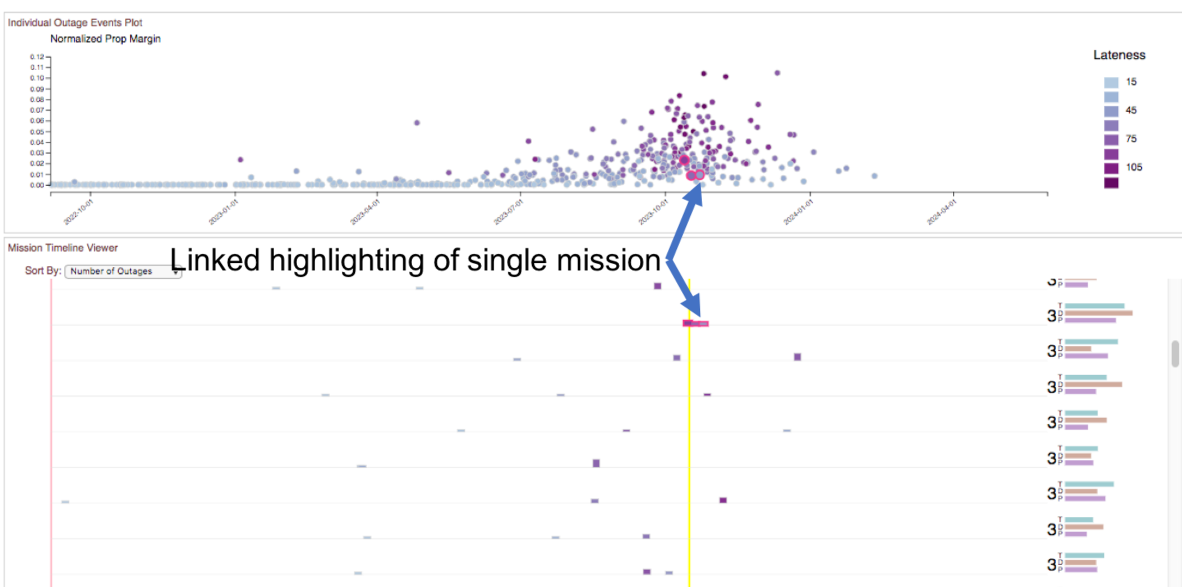


Fig. 7: Missed thrust Monte Carlo result interrogation environment, time line of outages for multiple samples as well as single-sample time lines (D3.js)

that this linked highlighting work when hovering over any individual data point, in any of the plots. The histograms in the upper right provide additional information about the distribution of lateness and propellant margin; the histograms are linked to the zoom feature such that they only display information from the center scatter. In the bottom of Fig. 6, a timeline scatter is shown where the outages of the zoomed sample are shown over time; the colors correspond to the lateness caused by an individual outage and the red circles indicate the outages of the specific sample run highlighted in the upper center plot.

Turning to Fig. 7, we again show the timeline scatter but now also include the timelines for individual Monte Carlo cases. As before, hovering over a data point highlights that specific sample for further investigation. Superimposed on the individual timelines is a yellow line indicating the nominal Mars arrival when no outages occur. As can be seen, outages near this nominal arrival condition have the most impact on the lateness and propellant margin consumed; outages early in the mission can easily be accommodated, but later outages have severe impacts. Included to the right of the individual timelines is a summary graphic providing summary information about that run, namely the time of engine outage, the delay in arrival, and the excess propellant consumed. In contrast to traditional text summaries and static plots, this visual environment allows for rapid exploration of the Monte Carlo output and enables the MDNav user to quickly focus attention on the most interesting regions for further analysis.

IV.iii Orbit Determination Dashboard

Our final MDNav task is the process of *Orbit Determination* for which we wish to enable an analyst (or team of analysts) the ability to rapidly compare different OD runs and intuitively link differences in inputs to variations in outputs. As a motivating use case, we simulate the operations of a fictitious Venus orbiting spacecraft dubbed "Aphrodite". Aphrodite is modeled to execute impulsive chemical maneuvers and move under the influence of Venus and solar gravity as well as solar radiation pressure. Measurements for filtering are provided solely by the Deep Space Network, specifically radiometric range and range-rate (Doppler).[36] The Monte software suite[8,9] is used to simulate OD for several revolutions around Venus; as is typical of an operations environment, several different runs are considered spanning a nominal case, varying *a priori* uncertainties on spacecraft state as well as maneuvers, data editing, treating gravitational parameters as "consider" variables in the filter, and estimating stochastic uncertainties on small forces.

For this application, we have created the Orbit Determination Dashboard (OD-D) using D3.js,[35] which allows comparing multiple *Inputs & Models* cases in a single tree-map[32] visualization, while showcasing differences in outputs in *Astrodynamic Plots*, namely residual viewers and uncertainty ellipsoids, on more standard plots with the addition of interactivity. Tree-maps represent a hierarchy by recursively subdividing a designated area as moving down through a hierarchy. Figure 8 presents the top-level tree structure of the simulation inputs, with different colors indicating specific logical groupings within the hierarchy, as indicated by the color key. In this overview mode, the size of the blocks represents the amount of data contained in each branch of the tree, with user defined scaling to emphasize blocks that are most relevant to the mission at hand (e.g., the red "Trajectory" and green "Filter" sets are provided an additional scaling factor because these factors are the most likely to affect the "operations" of Aphrodite).

To be able to compare multiple hierarchies, we first create a union hierarchy structure that contains every node in every hierarchy, with difference in values across are stored. In the resulting tree map, differences in hue are mapped to high level nodes in the hierarchy, nodes that show difference among any child node is highlighted with orange color icons in their bottom right corner. The visualization reveals two level depth of hierarchy at any given time. When the user clicks on a node, the animation scales that node to the whole area, and the children, and grandchildren of the selected node are visualized. The path or the depth of the selected node is displayed above the tree-map. Animation stops, when a terminal or "leaf" node is reached, as shown in Fig. 9. At this level, differences between the baseline and alternate analysis cases are presented both as summary text and, when appropriate, as plots. In this case, the bars indicate differences in the (x, y, z)-components of maneuver uncertainty for the first Orbit Trim Maneuver (OTM). Further text provides the user additional details about the OTM and variations in the input.

A set of prefit and postfit residual plotters as well as spacecraft state uncertainty ellipses are also included in OD-D, as shown in Fig. 10. Check boxes enable the operator to quickly toggle certain cases on and off in the plots, and interactive zoom features allow a user to focus on areas of particular concern.

Fig. 8: Orbit Determination Dashboard comparing inputs of multiple cases and indicating differences at the *top* level of the data tree (D3.js)

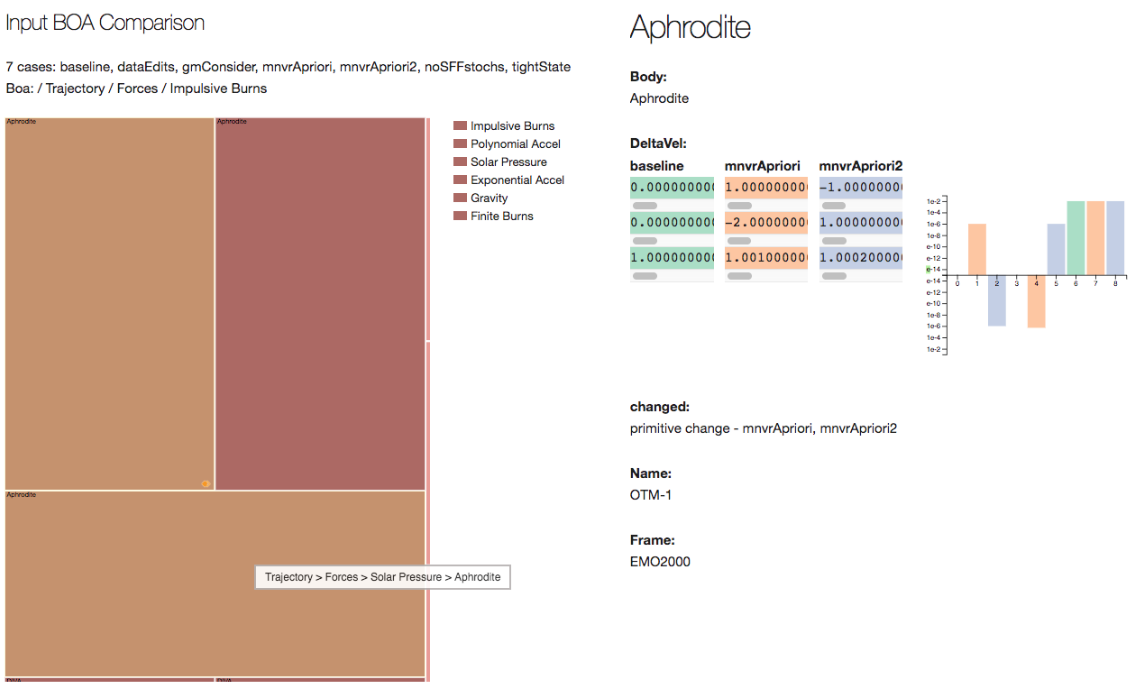


Fig. 9: Orbit Deterimination Dashboard (OD-D) comparing impulsive burn differences at the *bottom* level of the data tree (D3.js)

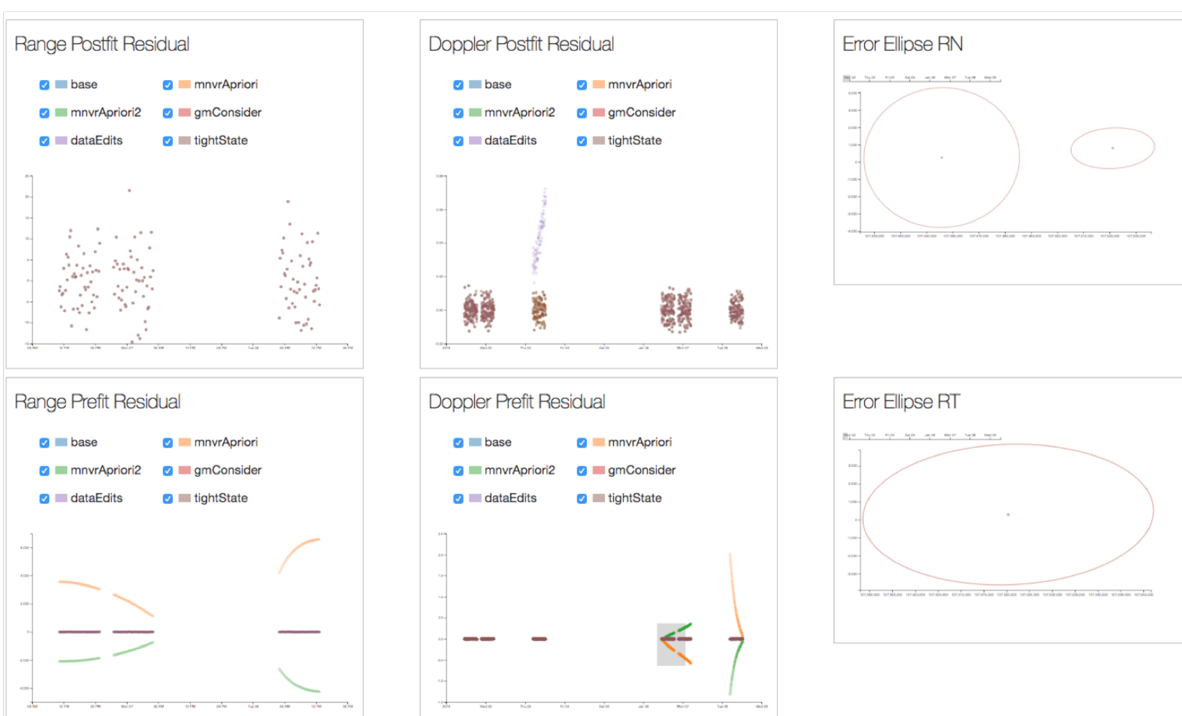


Fig. 10: Orbit Determination Dashboard plotting measurement residuals and uncertainty ellipses for multiple cases (D3.js)

Colors are used to denote the outputs of different runs. In OD-D's unified interactive display, individuals and teams of analysts can quickly assess the difference between runs, quickly formulate and check hypotheses, and determine the best run to use for future operations planning. Specific questions can be handled in-meeting, without the need for follow-up afterwards; key information about a run is stored in a logically consistent and compact manner, alleviating human cognitive effort to mentally catalogue and remember the differences between specific cases. This stands in stark contrast to current practice, wherein the outputs of multiple runs are compiled into a large slide deck requiring tedious scrolling back-and-forth with key distinguishing features of the runs encoded as sometimes abbreviated and obscure text labels.

V. Summary

We have completed a preliminary investigation into visual interactivity as applied to mission design and navigation. We have built prototype interactive work spaces to address specific tasks within early trajectory design, statistical contingency analysis, and orbit determination. While still in the preliminary stages of development, these initial models enable us to elicit feedback from the target end users and rapidly implement low-cost changes. This rapid iteration during software development is a key strategy for ultimate infusion into MDNav capabilities and ultimate adoption by mission operators.

Even in our initial study, several key advantages of interactive visualizations have already become apparent. First, interactivity reduces the time to discovery, often up to an order of magnitude or more; this increased performance improves the efficiency of MDNav teams and ultimately will lead to more comprehensive designs, more robust safety assurance, and improved operations. Second, interactivity encourages low-cost, low-risk exploration when implemented with the correct safeguards. Automatic or user-initiated tracking of design iterations as well as easy restores to a base state of analysis greatly alleviates the risk of losing valuable designs or key insights. Finally, visual environments benefit from a focus in scope: while visualization libraries or platforms can be comprehensive, specific applications should address specific tasks as efficiently as possible for the user. This library or "building block" paradigm enables a wide variety of missions to make use of the same fundamental analysis tools while enabling the customization needed for deep space exploration. This approach also helps to reduce clutter in the visual environment: if a particular mission doesn't need a certain visualization or widget, said object should be hidden from the main analysis work space. In general, mission teams should be able to customize the environment to suit their specific needs, elevating and suppressing certain features as desired. Software development should be scoped appropriately to enable this interlinking of different visualization instances with appropriate back-end analysis engines.

Many avenues remain open for future development, not the least being the continued iteration of the presented visualization approaches. Feedback from a variety of users across different projects will help to generalize these initial prototypes into more complete multi-mission visual interfaces. Additional investigation should be conducted into MDNav tasks from the mission life cycle that have not been covered to date. Certain useful concepts from the cases considered so far could also be applicable to other problems; for example, the Monte Carlo analysis tool could also be applied to trade space exploration. As more work tasks are considered, cross-cutting challenges can be identified and approached from a higher-level perspective. Additionally, application programming interfaces should begin to be formulated, enabling a standardized approach to development, more manageable infusion into existing software suites, and long-term stability and maintenance. Finally, the potential connections between MDNav tasks and visualization approaches deserves more thorough consideration; this investigation serves as a useful starting point, but as development continues currently unknown connections may reveal themselves.

VI. Acknowledgements

This research was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. The authors would also like to acknowledge the support of Brent Buffington from the Europa Clipper mission, Scott Davidoff from the Human-Computer Interfaces group, and the SunRISE mission proposal team. In addition to the authors, significant work was done by the following student interns: James Crowley, Michael Gutensohn, Andrew Cox, Shixie Tropimov, and Peter Polack.

References

[1] Museth, K., Barr, A., and Lo, M., "Semi-Immersive Space Mission Design and Visualization: Case Study of the 'Terrestrial Planet Finder' Mission", *Proceedings of the Con-*

ference on Visualization '01, San Diego, California, USA, October 21-26, 2001, pp. 501-504.

[2] Lee, K., and Trivailo, P., "Advanced Interactive Visualization and Simulation of Spacecraft Formation using Virtual Reality", 59^{th} *International Astronautical Congress*, Glasgow, United Kingdom, September 29 - October 3, 2008, IAC-08.E1.4.11.

[3] AGI, "Systems Tool Kit (STK)", retrieved on September 3, 2018, `http://www.agi.com/products/engineering-tools` and `https://p.widencdn.net/l1oj4s/STK-Product-Specsheet`.

[4] A.I. Solutions, "FreeFlyer", retrieved on September 3, 2018, `https://ai-solutions.com/freeflyer/`.

[5] NASA Jet Propulsion Laboratory, "SPICE-enhanced Cosmographia Mission Visualization Tool", retrieved on September 3, 2018, `https://naif.jpl.nasa.gov/naif/cosmographia.html`.

[6] NASA Goddard Space Flight Center, "General Mission Analysis Tool (GMAT)", retrieved on September 3, 2018, `https://software.nasa.gov/software/GSC-17177-1` and `http://gmatcentral.org/display/GW/GMAT+Wiki+Home`.

[7] NASA Johnson Space Center, "Copernicus Trajectory Design and Optimization System", retrieved on September 3, 2018, `https://software.nasa.gov/software/MSC-25863-1` and `https://www.nasa.gov/centers/johnson/copernicus/index.html`.

[8] NASA Jet Propulsion Laboratory, "Monte: Mission Analysis, Operations, and Navigation Toolkit Environment", retrieved on September 3, 2018, `https://montepy.jpl.nasa.gov/`.

[9] Evans, S., Taber, W., Drain, T., Smith, J., Wu., H-C., Guevara, M., Sunseri, R., and Evans, J., "Monte: The Next Generation of Mission Design & Navigation Software", *The* 6^{th} *International Conference on Astrodynamics Tools and Techniques (ICATT)*, Darmstadt, Germany, March 14-17, 2016, `https://indico.esa.int/indico/event/111/session/30/contribution/177`.

[10] Schlei, W. and Howell, K., "A Visual Analytics Approach to Preliminary Trajectory Design", *AAS/AIAA Space Flight Mechanics Meeting*, Charleston, South Carolina, January 29-February 2, 2012, AAS 12-190.

[11] Schlei, W., Howell, K., Tricoche, X., and Garth, C., "Enhanced Visualization and Autonomous Extraction of Poincar Topology", *Journal of the Astronautical Sciences*, Vol. 61, No. 2, June 2014, pp. 170-197 `https://dx.doi.org/10.1007/s40295-015-0042-4`.

[12] Schlei, W., *Interactive Spacecraft Trajectory Design Strategies featuring Poincaré Map Topology*, Ph.D. Dissertation, Purdue University, 2017, `https://engineering.purdue.edu/people/kathleen.howell.1/Publications/Dissertations/2017_Schlei.pdf`.

[13] Folta, D., Bosanac, N., Guzzetti, D., and Howell, K., "An Earth-Moon System Trajectory Design Reference Catalog", *Acta Astronautica*, Vol. 110, May-June 2015, pp. 341-353, `https://doi.org/10.1016/j.actaastro.2014.07.037`.

[14] Guzzetti, D., Bosanac, N., Haapala, A., Howell, K., Folta, D., "Rapid Trajectory Design in the Earth-Moon Ephemeris System via an Interactive Catalog of Periodic and Quasi-Periodic Orbits", *Acta Astronautica*, Vol. 126, September-October 2016, pp. 439-455, `https://dx.doi.org/10.1016/j.actaastro.2016.06.029`.

[15] Haapala, A., and Howell, K., "Representations of Higher-Dimensional Poincar Maps with Application to Spacecraft Trajectory Design", *Acta Astronautica*, Vol. 96, March-April 2014, pp. 23-41, `https://dx.doi.org/10.1016/j.actaastro.2013.11.019`.

[16] Craig Davis, D., Phillips, S., McCarthy, B., "Trajectory Design for Saturnian Ocean Worlds Orbiters using Multidimensional Poincaré Maps", *Acta Astronautica*, Vol. 143, February 2018, pp. 16-28, `https://doi.org/10.1016/j.actaastro.2017.11.004`.

[17] Buffington, B., Lam, T., and Campagnola, S., "Evolution of Trajectory Design Requirements on NASA's Planned Europa Clipper Mission", 68^{th} *International Astronautical Congress*, Adelaide, Australia, September 25-29, 2017, IAC-17.C1.7.8.

[18] Santos, F., de Almeida, A., and Krone-Martins, A., "Interactive 3D Visualization of Large Astronomical and Space Debris Datasets", 64^{th} *International Astronautical Congress*, Beijing, China, September 23-27, 2013, IAC-13,A6,1,10,x17592.

[19] Vrotsou, K., Janetzko, H., Navarra, C., Fuchs, G., Spretke, D., Mansmann, F., Andrienko, N., and Andrienko, G., "SimpliFly: A Methodology for Simplification and Thematic Enhancement of Trajectories", *IEEE Transactions on Visualization and Computer Graphics*, Vol. 21, No. 1, January 2015, pp. 107-121.

[20] Amini, F., Rufiange, S., Hossain, Z., Ventura, Q., Irani, P., and McGuffin, M., "The Impact of Interactivity on Comprehending 2D and 3D Visualizations of Movement Data", *IEEE Transactions on Visualization and Computer Graphics*, Vol. 21, No. 1, January 2015, pp. 122-135.

[21] NASA, *NASA Space Flight Program & Project Management Handbook*, September, 2014, pp. 2:9, `https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20150000400.pdf`.

[22] Stuart, J., McElrath, T., and Petropoulos, A., "Mission Design Analysis for the Martian Moon Phobos: Close Flybys, Missed Thrusts, and other In-Flight Entertainment", *AAS/AIAA Astrodynamics Specialist Conference*, Vail, Colorado, USA, August 9-13, 2015, AAS 15-756.

[23] Card, S., Mackinley, H., "Readings in information visualization: using vision to think", *Morgan Kaufmann*, 1999.

[24] Liu, S., Cui, W., Wu, Y., and Liu, M., "A survey on information visualization: recent advances and challenges",*The Visual Computer*, Springer, Vol. 30, No. 12, 2014, pgs. 1373-1393.

[25] Unity Technologies, "Unity", retrieved on September 3, 2018, `https://unity3d.com/`.

[26] Jordan, J., *The Application of Lambert's Theorem to the Solution of Interplanetary Transfer Problems*, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, USA, February 1, 1964, Technical Report No. 32-521.

[27] Riverbank Computing Limited, "PyQt", retrieved on September 3, 2018, `https://riverbankcomputing.com/software/pyqt/intro`.

[28] Stuart, J., Dorsey, A., Alibay, F., and Filipe, N., "Formation Flying and Position Determination for a Space-Based Interferometer in GEO Graveyard Orbit", *IEEE Aerospace Conference*, Big Sky, Montana, USA, March 4-11, 2017.

[29] Clohessy, W. and Wiltshire, R., "Terminal Guidance System for Satellite Rendezvous", *Journal of the Astronautical Sciences*, Vol. 27, No. 9, 1960, pgs.653-678.

[30] Barden, B. and Howell, K., "Fundamental Motions near Collinear Libration Points and Their Transitions", *Journal of the Astronautical Sciences*, Vol. 46, No. 4, 1998, pgs. 361-378.

[31] Hernandez, S., Stuart, J., Garza, D., Broschart, S., Herzig, S., and Chien, S., "Satellite Constellation Orbit Design to Enable a Space-Based Radio Interferometer", *AAS/AIAA Astrodynamics Specialist Conference*, Stevenson, Washington, USA, August 20-24, 2017, AAS 17-607.

[32] Johnson, B., and Shneiderman, B., "Tree-maps: A space-filling approach to the visualization of hierarchical information structures",*Proceedings of the IEEE Conference on Visualization'91*, IEEE, 1991, pgs. 284-291.

[33] Matplotlib Development Team,"matplotlib", retrieved on September 3, 2018, `https://matplotlib.org/`.

[34] Laipert, F. and Imken, T., "A Monte Carlo Approach to Measuring Trajectory Performance Subject to Missed Thrust", *AIAA SciTech Forum / AAS Space Flight Mechanics Meeting*, Kissimmee, Florida, USA, January 8-12, 2018, DOI: 10.2514/6.2018-0966.

[35] Bostock, M., "D3: Data-Driven Documents", retrieved on September 3, 2018, `https://d3js.org/` and `https://github.com/d3/d3`.

[36] Moyer, T., *Formulation for Observed and Computed Values of Deep Space Network Data Types for Navigation*, John Wiley & Sons, Inc., Hoboken, New Jersey, 2003.